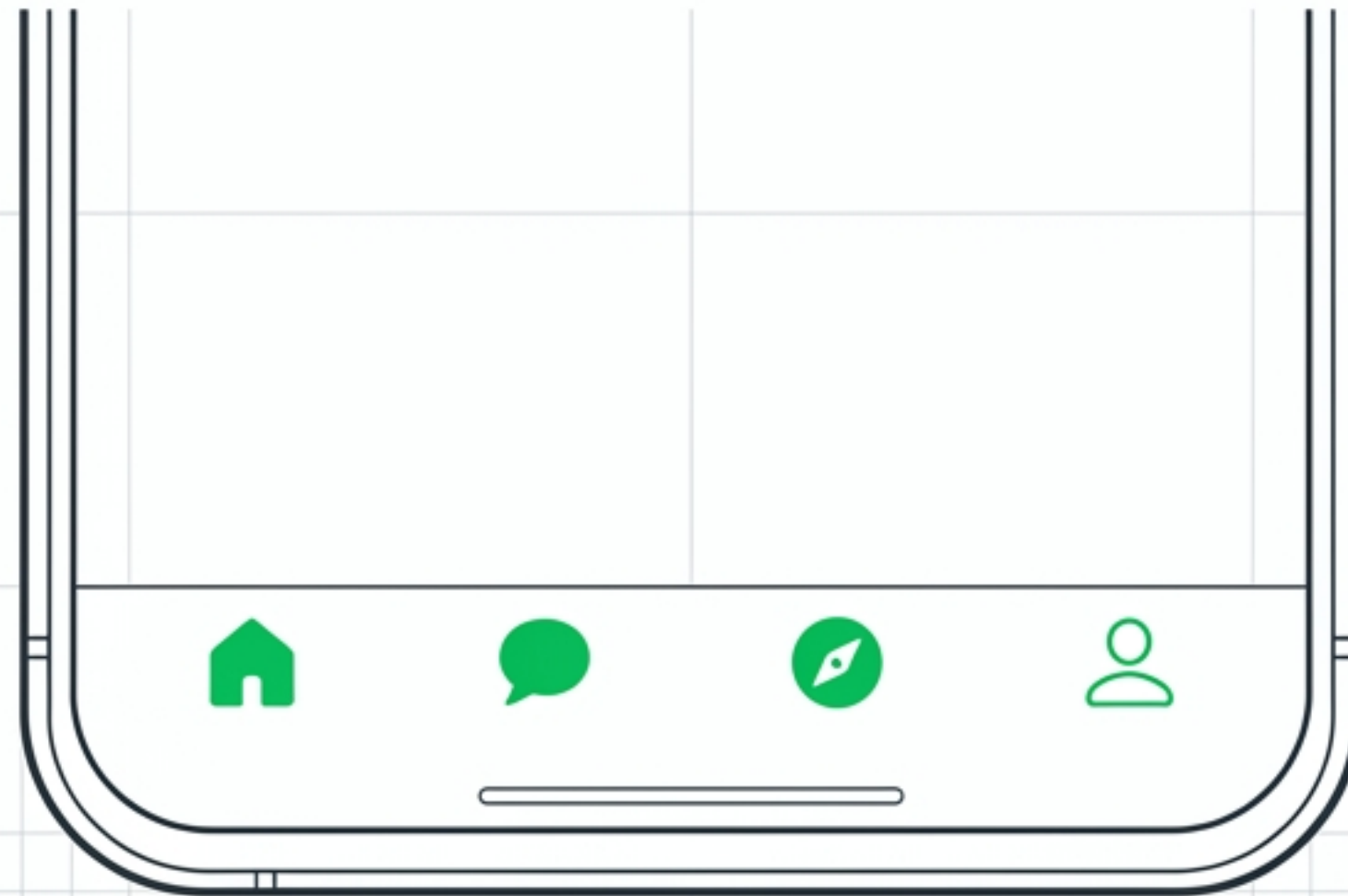


Building a Native Feel

The Mio v0.0.4 Rebuild: 30+ Components,
0 New Dependencies, 1 Familiar Mental Model.



Functional is not enough. Familiarity removes friction.

```
$
+ezint tesselas? {
  chat_response {
    oiaot:name: https://tepst-nont.com/goo/attarv? → {
      seased: "ons\neosertata"
    }
    alack:vame: https://tepst-oot.com/goo/athatv? → {
      seased: "dnd\neoverata"
    }
  }
  ...
  seased: "seased"
}
> |
```

The Terminal: Functional, but lacking a sense of place.



The App: Immediate context and navigation.

Mio was a production-ready Telegram bot, but Telegram is not where Chinese users live. The existing web app was a bare single-page chat. You open it, you type, you get a response.

What interface do **Chinese users already understand without any onboarding?**

Designing for 1.3 Billion Fingers



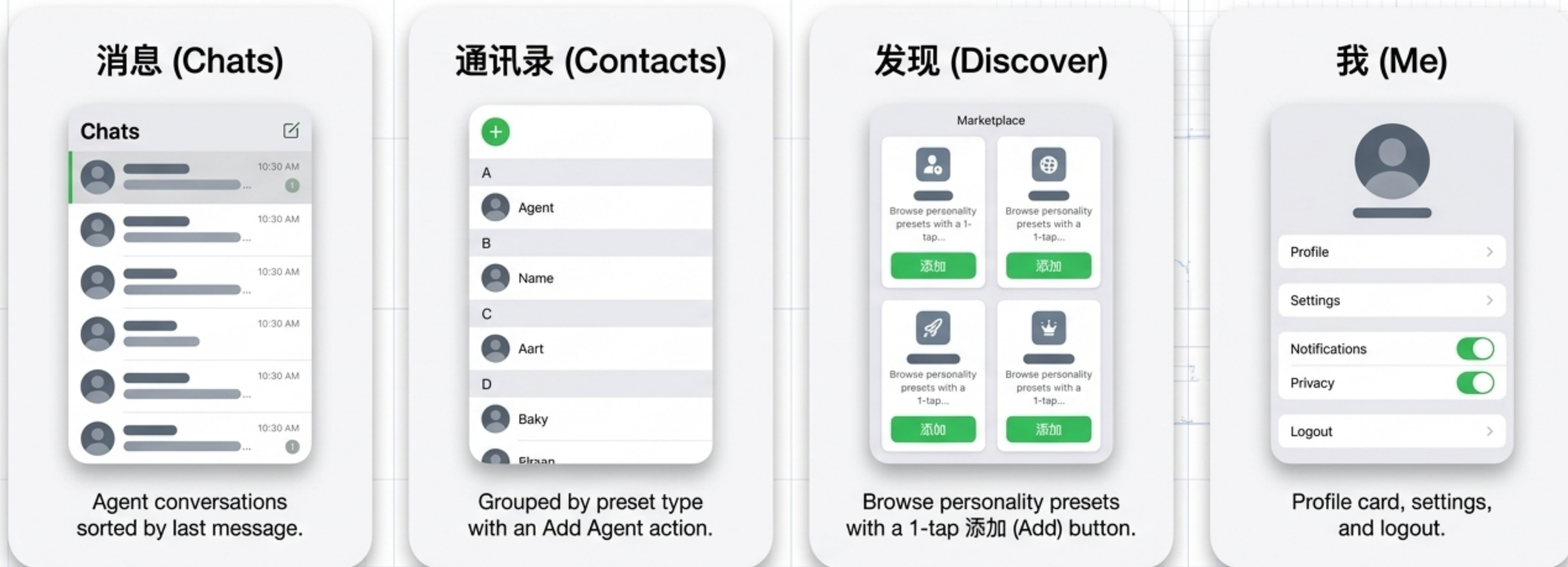
The interface was rebuilt around WeChat—not just as a UI pattern, but as deeply ingrained muscle memory.

A user opening the app for the first time should never have to think about how to use it. They should already know where everything is.

Key Insight Box

The best UI is one the user already learned somewhere else. It is not a UI pattern; it is a reflex.

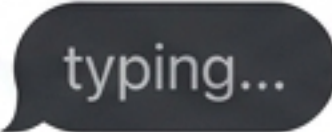

A Proper Multi-Page Architecture



Not a SPA pretending to have tabs. Each tab is a real page with its own route, protected by middleware.

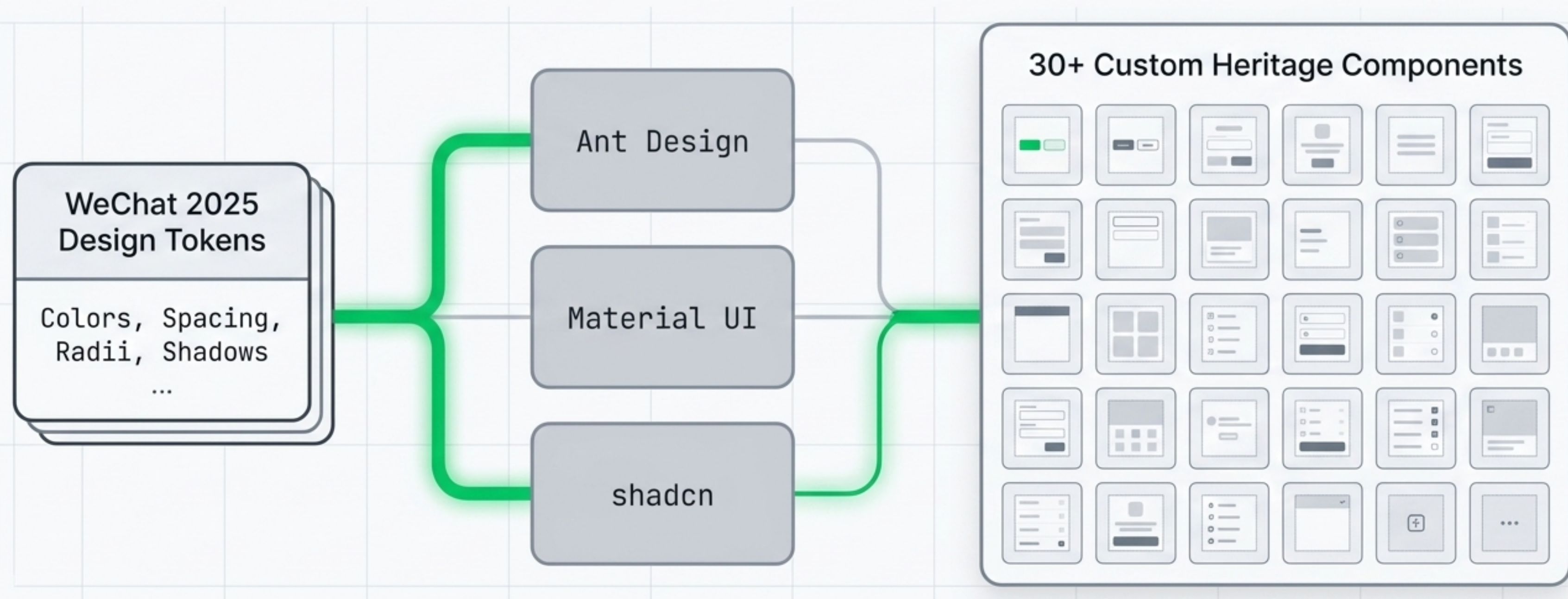
Native, Not Translated

Every piece of user-facing text is written in Chinese from line one. “对方正在输入...” carries a specific familiarity that no English translation can manufacture.

	Imported / Translated UX	Native / Chinese-First UX
The Typing Indicator		
Tone	Formal and stiff spacing.	Casual, exact strings seen thousands of times daily.
User Perception	Feels like a React app with Chinese text.	Feels like a familiar, lived-in ecosystem.

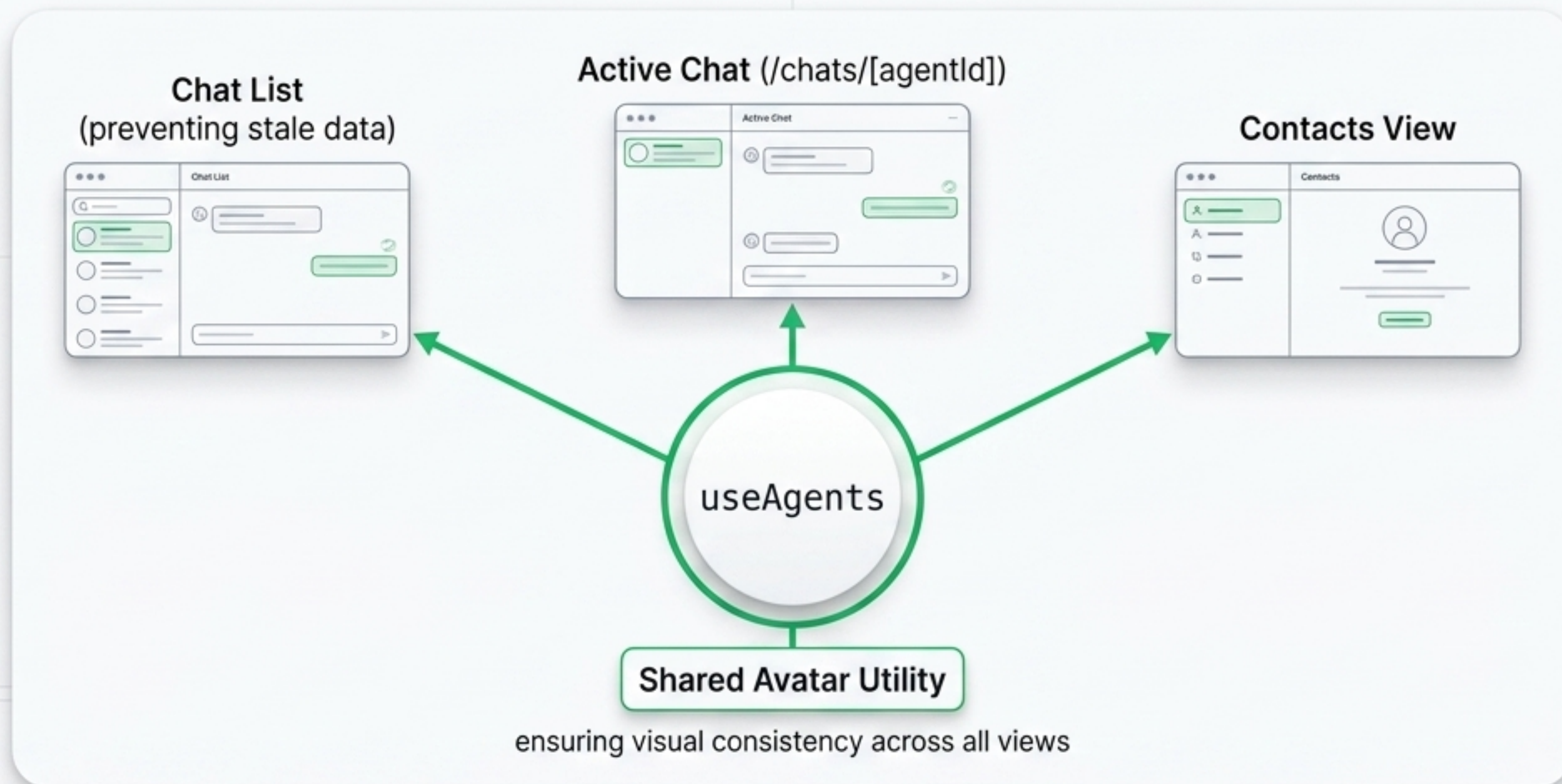
30 Components. 0 Dependencies.

Relying on external libraries introduces friction against a highly opinionated visual language. Building from scratch using CSS custom properties ensures absolute visual consistency without adding a single package to package.json.



One Hook, One Source of Truth

The useAgents hook manages the agent list and session data across all tabs in under 100 lines of code.



Erasing the Form Fill Experience

Onboarding sets the tone. If the first experience feels mechanical, users treat the product mechanically. Presenting backstory choices as chat bubbles means the user has their first conversation with Mio before onboarding is even finished.

Username
[Redacted]

Email
[Redacted]

Password
[Redacted]

Select Backstory
Select Backstory... ▾

Adventurer Merchant
 Scholar Warrior

Mechanical / Telegram Standard.

Hi there! Let's get you set up. What kind of backstory resonates with you?

Mio: We want to tailor the experience to your preferences.

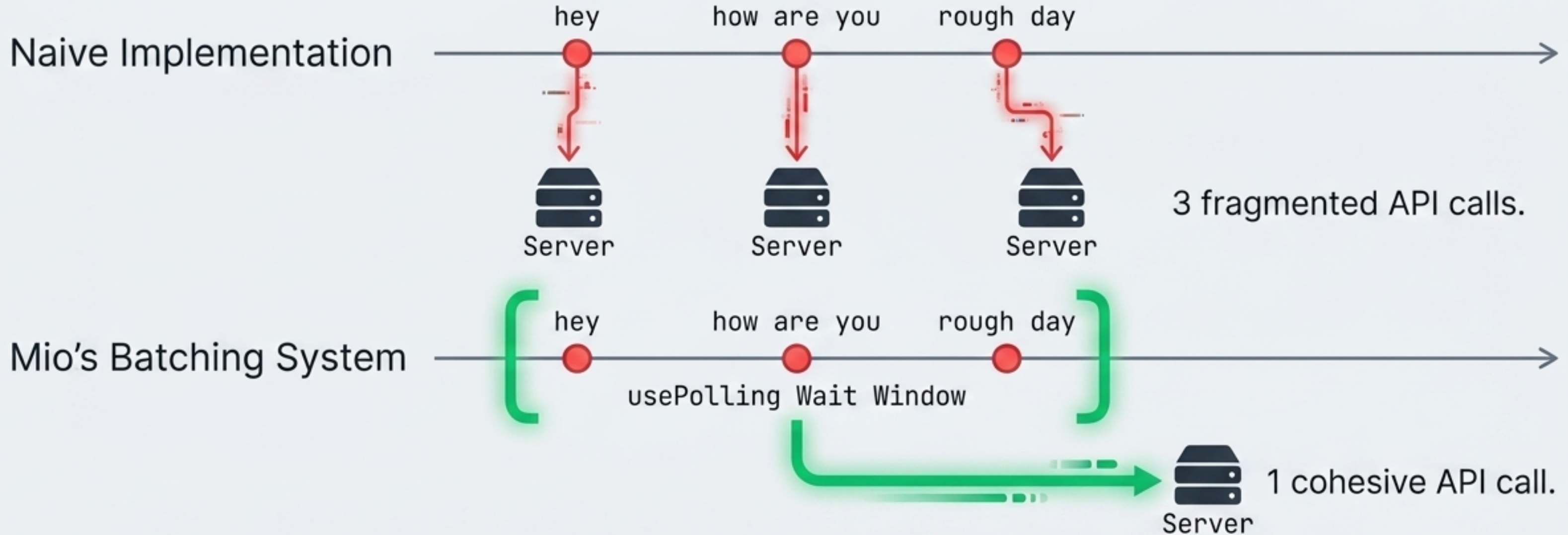
Adventurer at heart
A learned Scholar
Merchant with ambition.

Mio: Perfect! An Adventurer it is. Now, let's continue.

Conversational / Web Native

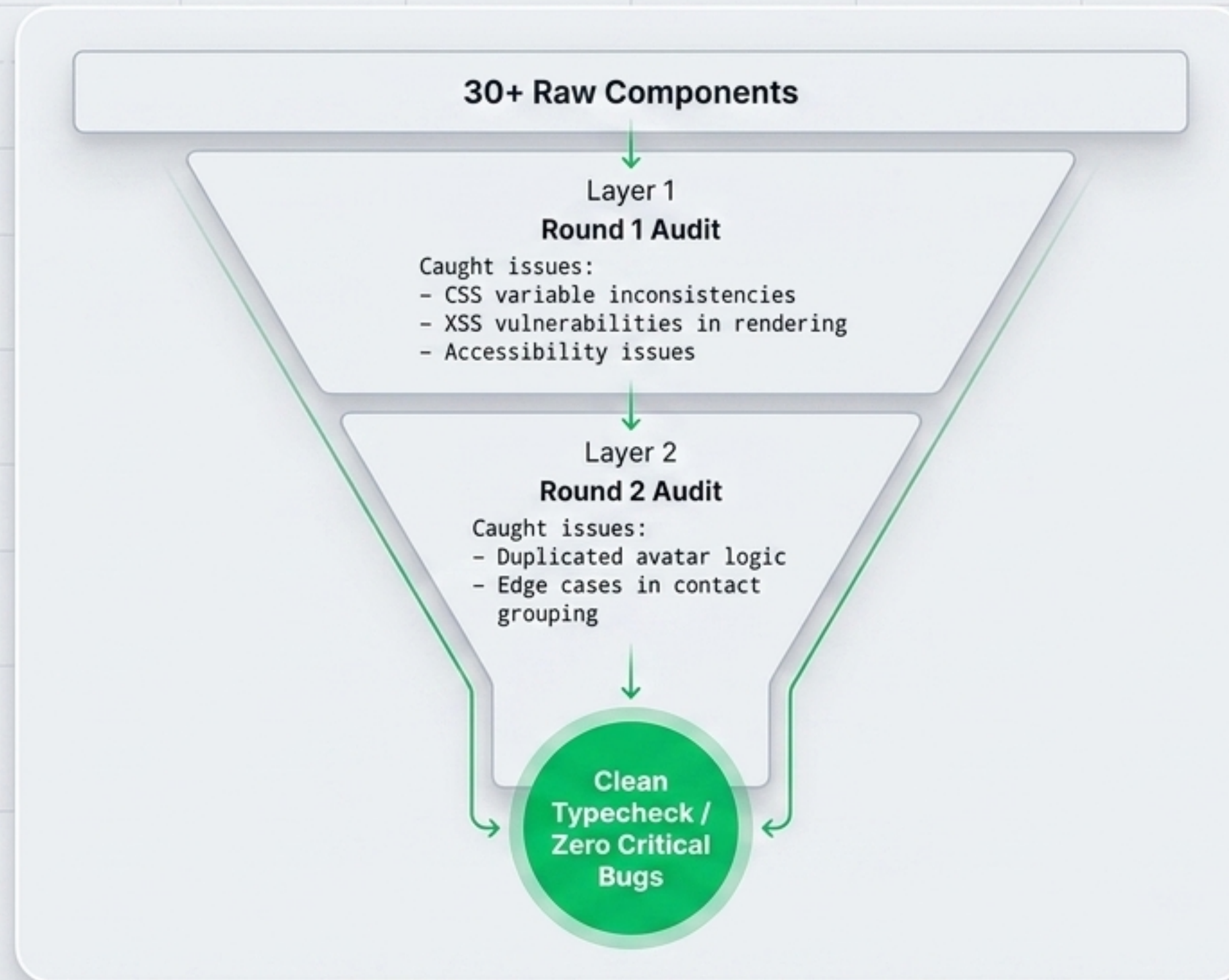
Mimicking Human Rhythms

Real people send rapid-fire short messages. Mio's client-side batching system watches typing patterns and groups rapid texts into single thoughts, perfectly mirroring actual chat behavior.



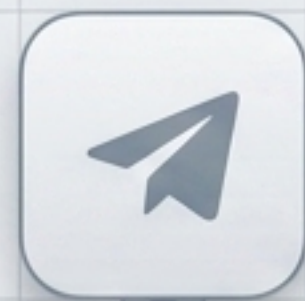
Ship When the Audit is Clean

A feature without an audit is a liability. Every CRITICAL and HIGH issue was addressed and avatar logic was centralized before release.



From Bot is Bot to Platform: Controlling the Experience

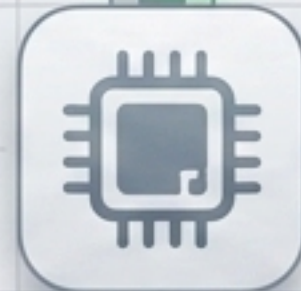
A Telegram bot lives inside someone else's rules. A web app is yours. Mio has evolved from an architectural experiment into a multi-platform product with its own design language.



The Telegram Interface.
Borrowed platform, constrained
UX, messaging-first users.



The Mio Web Platform.
Owned experience, custom onboarding,
full ecosystem control.



Mio Core AI Engine
(Personality & Memory)

The foundation is solid.
Next up (v0.0.5): Making Mio speak.